UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/560,839	06/08/2006	Elmar Thurner	2003P06167WOUS	6878

22116 7590 12/11/2007

SIEMENS CORPORATION
INTELLECTUAL PROPERTY DEPARTMENT
170 WOOD AVENUE SOUTH
ISELIN, NJ 08830

EXAMINER	
GAMI, TEJAL	

ART UNIT	PAPER NUMBER
2121	

MAIL DATE	DELIVERY MODE
12/11/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/560,839	THURNER, ELMAR	
	Examiner	Art Unit	
	Tejal J. Gami	2121	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 September 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *25,27-31,34-38,40,41 and 43-47* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *25,27-31,34-38,40,41 and 43-47* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This office action is responsive to an AMENDMENT entered September 28, 2007 for the patent application 10/560839.

Status of Claims

2. Claims 25-47 were rejected in the last Office Action dated July 02, 2007.

As a response to the July 02, 2007 office action, Applicant has Amended claims 25, 27, 28, 34, 35, 37, 38, 41, 44, 46 and 47; and Cancelled claims 26, 32, 33, 39, and 42.

Claims 25, 27-31, 34-38, 40, 41 and 43-47 are now pending in this office action.

Specification Objections

3. Examiner thanks Applicant for amending the abstract in response to the objections of the previous office action. Those objections have been withdrawn.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States

only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 25, 27-31, 34-38, 40, 41 and 43-47 are rejected under 35 U.S.C. 102(e) as being anticipated by Lo et al. (U.S. Publication Number 2003/0061349).

As to independent claim 25, Lo discloses a method for executing a program for an industrial automation system (see Paragraph [0001]), comprising:

providing a computer unit with input aids (see Paragraph [0046]), output aids and a display device (see Paragraph [0046]), having modules and functions respectively representing sub-tasks (e.g., employ graphical programming schemes) of an automation solution being modeled and/or created using the input aids and optionally the display device (see Paragraph [0047]), having the modules and functions (e.g., Function Block Diagrams FBD) (see Paragraph [0053]) being structured and networked using the input aids and optionally the display device as to form at least one hierarchical tree as a machine-independent program (see Paragraph [0048] where XML documents have the structure of hierarchical trees), loading the machine-independent program in the form of the at least one hierarchical tree into the corresponding components (e.g., PLC) (see Paragraph [0045]) of the automation system (e.g., download to a programmable controller) (see Paragraph [0057]), wherein the corresponding components of the automation system execute the or each machine-independent program present in the form of the at least one hierarchical tree (see Paragraph [0048] where XML documents have the structure of hierarchical trees) with the aid of at least one object machine assigned to the corresponding components (e.g., linked) of the automation system (e.g., PLC) (see Paragraph [0070]), and wherein the at least one

object machine provides operators and objects (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) from which the machine-independent program is provided in the form of the at least one hierarchical tree (see Paragraph [0048] where XML documents have the structure of hierarchical trees); and

during or after loading of the machine-independent program (e.g., download compiled code to a programmable controller) (see Figure 16, 1620), instantiating the operators (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) using the at least one object machine into corresponding components of the automation system (e.g. PLC) (see Paragraph [0070]); and

converting the symbolic representation of the hierarchical tree to physical addresses (e.g., mapping logical to physical I/O) (see Paragraph [0082]) to generate a loadable program (e.g., downloaded) (see Paragraph [0082]) in the form of an executable program or operator tree (e.g., execute code) (see Paragraph [0007]).

As to independent claim 38, Lo discloses a device for executing a program for an industrial automation system (see Paragraph [0001]), comprising:

at least one computer unit with input aids, output aids and a display device (see Paragraph [0046]);

a component for modeling and/or creating modules and functions (e.g., Function Block Diagrams FBD), which respectively represent the sub-tasks (e.g., employ graphical programming schemes) of an automation solution (see Paragraph [0047] and [0075]);

a component for structuring the modules and functions and for networking the same, to form at least one hierarchical tree as at least one machine-independent program (see Paragraph [0048] where XML documents have the structure of hierarchical trees); and

a component to load the or each machine-independent program (e.g., download to a programmable controller) (see Paragraph [0057]) in the form of at least one hierarchical tree into the corresponding components (e.g., PLC) (see Paragraph [0045]) of the automation system with the corresponding components of the automation system executing the machine-independent program present in the form of the at least one hierarchical tree (see Paragraph [0048] where XML documents have the structure of hierarchical trees), wherein at least one object machine is assigned to the corresponding components (e.g., linked) of the automation system (e.g., PLC) (see Paragraph [0070]) to execute the machine-independent programs (e.g., operating system independent) (see Paragraph [0047]-[0048]), and wherein the at least one object machine provides operators and objects (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) from which the machine-independent program is provided in the form of the hierarchical tree (see Paragraph [0048] where XML documents have the structure of hierarchical trees);

a component to instantiate the operators (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) using the at least one object machine during or after the loading of the machine-independent program into corresponding components of the automation system (e.g. PLC) (see Paragraph [0070]); and

a component to convert the symbolic representation of the at least one hierarchical tree to physical addresses (e.g., mapping logical to physical I/O) (see Paragraph [0082]) to generate a loadable program (e.g., downloaded) (see Paragraph [0082]) in the form of an executable program or operator tree (e.g., execute code) (see Paragraph [0007]).

As to independent claim 47, Lo discloses a computer program implementing a method for executing a program for an industrial automation system (see Abstract), comprising:

providing a computer unit with input aids, output aids and a display device (see Paragraph [0046]), having modules and functions (e.g., Function Block Diagrams FBD) respectively representing sub-tasks (e.g., employ graphical programming schemes) of an automation solution being modeled and/or created using the input aids and optionally the display device (see Paragraph [0047] and [0075]), having the modules and functions (e.g., Function Block Diagrams FBD) (see Paragraph [0053]) being structured and networked using the input aids and optionally the display device as to form a hierarchical tree as a machine-independent program (see Paragraph [0048] where XML documents have the structure of hierarchical trees), loading the machine-independent program in the form of the hierarchical tree into the corresponding components (e.g., PLC) (see Paragraph [0045]) of the automation system (e.g., download to a programmable controller) (see Paragraph [0057]), wherein the corresponding components of the automation system execute the or each machine-independent program present in the form of the hierarchical tree (see Paragraph [0048] where XML

documents have the structure of hierarchical trees) with the aid of at least one object machine assigned to the corresponding components (e.g., linked) of the automation system (e.g., PLC) (see Paragraph [0070]), and wherein the at least one object machine provides operators and objects (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) from which the machine-independent program is provided in the form of the hierarchical tree (see Paragraph [0048] where XML documents have the structure of hierarchical trees); and

during or after loading of the machine-independent program (e.g., download compiled code to a programmable controller) (see Figure 16, 1620), instantiating the operators (e.g., operations or functions) (see Figure 1: AND OR; and Paragraph [0004]) using the at least one object machine into corresponding components of the automation system (e.g. PLC) (see Paragraph [0070]); and

converting the symbolic representation of the hierarchical tree to physical addresses (e.g., mapping logical to physical I/O) (see Paragraph [0082]) to generate a loadable program (e.g., downloaded) (see Paragraph [0082]) in the form of an executable program or operator tree (e.g., execute code) (see Paragraph [0007]).

As to dependent claim 27, Lo teaches the method according to claim 25, wherein the machine-independent program is present in the form of at least one hierarchical object or operator tree in the corresponding components of the automation system and are processed interpretatively (see Paragraph [0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 28, Lo teaches the method according to claim 27, wherein the machine-independent program is present in the form of at least one object or operator tree with a structure equivalent or similar to the representation of the program in the or each display device (see Paragraph [0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 29, Lo teaches the method according to claim 25, wherein the machine-independent program is loaded into the corresponding components of the automation system using a machine-independent, symbolic representation of the hierarchical tree (e.g., symbolic designation) (see Paragraph [0053]).

As to dependent claim 30, Lo teaches the method according to claim 29, wherein the machine-independent and symbolic representation of the hierarchical tree is in the form of a byte code or a markup language such as extended markup language (e.g., XML) (see Paragraph [0048]).

As to dependent claim 31, Lo teaches the method according to claim 25, wherein the object machine is configured as a real-time object machine with deterministic response and cycle times (e.g., cycle time) (see Paragraph [0007]).

As to dependent claim 34, Lo teaches the method according to claim 25, wherein the object machine is implemented as a function unit that is closed and that processes the hierarchical tree to a runtime (see Paragraphs [0047]-[0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 35, Lo teaches the method according to claim 27, wherein the object machine is implemented in a distributed manner as at least one object, with the hierarchical object or operator tree processing itself (see Paragraph [0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 36, Lo teaches the method according to claim 25, wherein the modules and functions (e.g., Function Block Diagrams FBD) are assigned model information and/or meta-information using the input aids and/or the display device (see Paragraphs [0046]-[0047]).

As to dependent claim 37, Lo teaches the method according to claim 27, wherein the objects of the machine-independent program present as a hierarchical object or operator tree are assigned a collection of infrastructure services or infrastructure functions that access the objects via containers assigned to the objects such that an infrastructure service or an infrastructure function can be used by all the objects (e.g., ActiveX) (see Paragraph [0064]).

As to dependent claim 40, Lo teaches the device according to claim 38, wherein the machine-independent program is present in the form of at least one object or operator tree with a structure equivalent or similar to the representation of the program in the display device (see Paragraph [0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 41, Lo teaches the device according to claim 38, wherein the object machine is configured as a real-time object machine with deterministic response and cycle times (e.g., cycle time) (see Paragraph [0007]).

As to dependent claim 43, Lo teaches the device according to claim 38, further comprising a device for assigning model information and/or meta-information to the modules and functions (e.g., tag) (see Paragraph [0053] and [0070]).

As to dependent claim 44, Lo teaches the device according to claim 38, wherein the object machine is implemented as a function unit that is closed and processes the or each hierarchical tree to a runtime (see Paragraphs [0047]-[0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 45, Lo teaches the device according to claim 38, wherein the object machine is implemented in a distributed manner as at least one object, with the hierarchical object or operator tree processing itself (see Paragraph [0048] where XML documents have the structure of hierarchical trees).

As to dependent claim 46, Lo teaches the device according to claim 38, wherein the objects of the machine-independent program present as a hierarchical object or operator tree are assigned a collection of infrastructure services or infrastructure functions that access the objects via containers assigned to the objects such that an infrastructure service or infrastructure function can be used by all the objects (see Paragraph [0053] for tools support all programming languages; and Paragraph [0070] for robust communication and data transfer from a shop floor to a board room).

Response to Arguments

6. Applicant's amendment and arguments filed September 28, 2007

have been fully considered. The amendment does not overcome the original art rejection and the arguments are not persuasive. The following are the Examiner's observations in regard thereto.

Applicant Argues:

Upon review and consideration of the Lo reference as a whole, Lo is wholly silent as to at least the elements: "during or after loading of the machine-independent program, instantiating the operators using the at least one object machine into corresponding components of the automation system"; and "converting the symbolic representation of the hierarchical tree to physical addresses to generate a loadable program in the form of an executable program or operator tree."

Examiner Responds:

Examiner is not persuaded. The newly added limitations to independent claims 25, 38, and 47 are anticipated by the prior art. See office action above for prior art teachings of claim limitations.

Applicant Argues:

Claims 34 and 44 each require that "the object machine is implemented as a function unit that is closed and that processes the hierarchical tree to a runtime." Claims 34 and 44 provide further reason for allowance because Lo, including paragraphs [0047]-[0048] cited by the Examiner, is wholly silent as to an object machine that is implemented as a function unit that is closed and that processes a hierarchical tree to a runtime.

Examiner Responds:

Examiner is not persuaded. The claims and only the claims form the metes and bounds of the invention. See Paragraphs [0047] and [0048] where Lo teaches running in an application making them operating system independent. Under such considerations, the prior art teaches the claims as written.

Applicant Argues:

Claims 37 and 46 require that the objects of the machine-independent program present as a hierarchical object or operator tree are assigned a collection of infrastructure services or infrastructure functions that access the objects via containers assigned to the objects such that an infrastructure service or an infrastructure function can be used by all the objects. As set forth in paragraph [0033], by so doing, "[i]nfrastructure services or corresponding functions are accessed via the container 32 and such access is the same for all objects in the hierarchical tree." Lo is wholly silent as to any containers, and moreover containers assigned to objects such that an infrastructure service or an infrastructure function can be used by all the objects.

Examiner Responds:

Examiner is not persuaded. See Paragraph [0064] where Lo teaches ActiveX containers. It is well known in the art at the time the invention was made that ActiveX is Microsoft technology used for developing reusable object oriented software components. Under such consideration, containers is taught by the prior art.

Conclusion

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Baier et al. (U.S. Patent Number: 7,151,966) teaches system and methodology providing open interface and distributed processing in an industrial controller environment.

Bloch et al. (U.S. Publication Number: 2002/0129129) teaches system and method for deploying and implementing software applications over a distributed network.

Langkafel et al. (U.S. Publication Number: 2005/0108265) teaches system and method for projecting transformations of object trees.

Klindt et al. (U.S. Patent Number: 6,853,867) teaches interface to a programmable logic controller.

Thurner (U.S. Publication Number: 2003/0009572) teaches system, method and apparatus of providing process data to a client.

8. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tejal J. Gami whose telephone number is (571) 270-1035. The examiner can normally be reached on Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David Vincent can be reached on (571) 272-3080. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

David Vincent
Supervisory Patent Examiner
Tech Center 2100

TJG

TJG

RONALD HARTMAN, JR.
PRIMARY EXAMINER
12/7/2007